PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G06F 13/00, 13/38, 17/00, 17/16, 17/21 — A1

(11) International Publication Number: **WO 00/26795**

(43) International Publication Date: 11 May 2000 (11.05.00)

(21) International Application Number: PCT/US99/24359

(22) International Filing Date: 18 October 1999 (18.10.99)

(30) Priority Data:
09/183,871 30 October 1998 (30.10.98) US

(71) Applicant: JUSTSYSTEM PITTSBURGH RESEARCH CENTER, INC. [US/US]; 4616 Henry Street, Pittsburgh, PA 15213 (US).

(72) Inventors: KANTROWITZ, Mark; 5503 Covode Street, Pittsburgh, PA 15217 (US). MCCALLUM, Andrew; 6623 Dalzell Place, Pittsburgh, PA 15217 (US). BERNSTEIN, Evan; 10 Lancaster Road, Freehold, NJ 07728 (US).

(74) Agents: BYRNE, Richard, L. et al.; Webb Ziesenheim Logsdon Orkin & Hanson, P.C., 700 Koppers Building, 436 Seventh Avenue, Pittsburgh, PA 15219–1818 (US).

(81) Designated States: AE, AL, AM, AT, AT (Utility model), AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, CZ (Utility model), DE, DE (Utility model), DK, DK (Utility model), DM, EE, EE (Utility model), ES, FI, FI (Utility model), GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SK (Utility model), SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published
With international search report.

(54) Title: METHOD FOR CONTENT–BASED FILTERING OF MESSAGES BY ANALYZING TERM CHARACTERISTICS WITHIN A MESSAGE

(57) Abstract

A computer implemented method for document classification or filtering of junk messages comprises the steps of computing the sum of the product of the frequency of occurrence with an assigned term weight for every term from a term lexicon that also appears in the message, normalizing the resulting sum by dividing the result by the total number of words (or the number of unique words) in the document and assigning a score to the document based on the normalized sum.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	**ES**	Spain	**LS**	Lesotho	**SI**	Slovenia
AM	Armenia	**FI**	Finland	**LT**	Lithuania	**SK**	Slovakia
AT	Austria	**FR**	France	**LU**	Luxembourg	**SN**	Senegal
AU	Australia	**GA**	Gabon	**LV**	Latvia	**SZ**	Swaziland
AZ	Azerbaijan	**GB**	United Kingdom	**MC**	Monaco	**TD**	Chad
BA	Bosnia and Herzegovina	**GE**	Georgia	**MD**	Republic of Moldova	**TG**	Togo
BB	Barbados	**GH**	Ghana	**MG**	Madagascar	**TJ**	Tajikistan
BE	Belgium	**GN**	Guinea	**MK**	The former Yugoslav Republic of Macedonia	**TM**	Turkmenistan
BF	Burkina Faso	**GR**	Greece			**TR**	Turkey
BG	Bulgaria	**HU**	Hungary	**ML**	Mali	**TT**	Trinidad and Tobago
BJ	Benin	**IE**	Ireland	**MN**	Mongolia	**UA**	Ukraine
BR	Brazil	**IL**	Israel	**MR**	Mauritania	**UG**	Uganda
BY	Belarus	**IS**	Iceland	**MW**	Malawi	**US**	United States of America
CA	Canada	**IT**	Italy	**MX**	Mexico	**UZ**	Uzbekistan
CF	Central African Republic	**JP**	Japan	**NE**	Niger	**VN**	Viet Nam
CG	Congo	**KE**	Kenya	**NL**	Netherlands	**YU**	Yugoslavia
CH	Switzerland	**KG**	Kyrgyzstan	**NO**	Norway	**ZW**	Zimbabwe
CI	Côte d'Ivoire	**KP**	Democratic People's Republic of Korea	**NZ**	New Zealand		
CM	Cameroon			**PL**	Poland		
CN	China	**KR**	Republic of Korea	**PT**	Portugal		
CU	Cuba	**KZ**	Kazakstan	**RO**	Romania		
CZ	Czech Republic	**LC**	Saint Lucia	**RU**	Russian Federation		
DE	Germany	**LI**	Liechtenstein	**SD**	Sudan		
DK	Denmark	**LK**	Sri Lanka	**SE**	Sweden		
EE	Estonia	**LR**	Liberia	**SG**	Singapore		

METHOD FOR CONTENT-BASED FILTERING OF MESSAGES BY ANALYZING TERM CHARACTERISTICS WITHIN A MESSAGE

BACKGROUND OF THE INVENTION

In this patent, the term "junk messages" is used to refer to both junk e-mail messages and junk newsgroup messages.

Junk messages represent a major and growing problem for the Internet and World Wide Web. Junk messages include many types of messages that the recipient does not wish to read, including messages containing unsolicited commercial advertisements, chain letters, scams and frauds, such as multi-level marketing schemes and get-rich-quick schemes, advertisements for adult services and spam. (Spam is a vernacular term for messages that are posted to an excessive number of newsgroups.)

Junk messages are harmful because they shift the burden of determining importance from sender to recipient, externalizing the true costs of the junk. The sender has no direct incentive to consider the wishes of the recipient.

Junk messages waste the recipient's time and money. It takes time to download, identify and discard the junk messages. This buries important messages, causing a loss of productivity. If the recipient pays for connect time and telephone calls, the junk messages cost the recipient money, akin to postage due advertisements. On flat-rate dial up services, the service provider pays for the junk messages in terms of wasted bandwidth and disk space. These costs are ultimately passed on to the recipient. The problem will continue to grow as more people become connected to the Internet.

Most current methods for filtering out junk messages use the headers of the message to identify the junk mail. These programs maintain extensive blacklists of the e-mail addresses, domain names and IP addresses of sources of junk messages and remove any messages from those sources. They may also filter based on other header fields (e.g., peculiarities in the recipient address) or the telltale signs of forged message headers. Comparing two of the largest blacklists with a large corpus of junk messages

found that this method identifies only about 70% of the junk messages.

Another popular method is to filter messages which were transmitted via blind carbon copy or a mailing list. Such messages can be easily identified because the recipient's address does not appear in the recipient fields of the header; but then the recipient must maintain a whitelist of legitimate sources of mail, such as his or her mailing list subscriptions and the e-mail addresses of colleagues who might send a message via blind carbon copy, to avoid filtering out legitimate messages. This heuristic would have caught only about 50% of the junk messages in our corpus.

To summarize, a blacklist is a list of header specifiers used to block messages and a whitelist is a list of header specifiers used to allow messages which would otherwise be filtered out to pass through the blockade.

Unfortunately, blacklists have many problems. They must be constantly updated as the large-scale offenders frequently change domain names and forge return addresses. Many junk messages come from first-time offenders and hence cannot be detected using a blacklist. The offender can also address the messages individually with randomly selected forged return addresses. Header based methods also cannot detect messages transmitted via a mailing list to which the recipient subscribes, nor junk messages posted to newsgroups. The provider of a blacklist faces the possibility of litigation for defamation and restraint of trade, especially if legitimate users and domains are accidentally or intentionally included in the blacklist.

DESCRIPTION OF THE PRIOR ART

W. Tietz, Electronic delivery of unwanted messages in open communications systems, NTZ (Germany), 47(2):74-7, February 1994.

Cynthia Dwork and Moni Naor, Pricing via processing or combating Junk Mail, Weizmann Institute of

Science, Department of Applied Mathematics and Computer Science, Technical Report CS95-20, 1995.

Douglas W. Oard and Gary Marchionini, A Conceptual Framework for Text Filtering, University of Maryland at College Park, Technical Report CS-TR-3643, May 1996.

Jason Rennie, ifile mail filtering system, http://www.cs.cmu.edu/~jr6b/ifile/ifile.

U.S. Patent No. 5,619,648 entitled "Message Filtering Techniques", Lucent Technologies Inc., filed November 30, 1994, issued April 8, 1997.

U.S. Patent No. 5,283,856 entitled "Event-Driven Rule-Based Messaging System", Beyond Inc., filed October 4, 1991, issued February 1, 1994. See also related U.S. Patent No. 5,555,346.

U.S. Patent No. 5,627,764 entitled "Automatic Electronic Messaging System With Feedback and Work Flow Administration", Banyan Systems, Inc., filed June 9, 1993, issued May 6, 1997.

U.S. Patent No. 5,377,354 entitled "Method and System for Sorting and Prioritizing Electronic Mail Messages", Digital Equipment Corporation, filed June 8, 1993, issued December 27, 1994.

There are numerous patents dealing with variations on the TFIDF method, including U.S. Patents Nos. 5,576,954; 5,659,766; 5,687,364; 5,371,807; and 5,675,819. The TFIDF computes the ratio of the frequency of each term in a document (TF) with the percentage of documents in which the term appears (IDF). IDF stands for inverse term frequency.

TFIDF uses IDF to emphasize terms which occur frequently in the document but relatively rarely in the collection of documents. In contrast, TDTF disclosed herein tries to emphasize terms which occur frequently in the message and which are good indicators of junk messages (i.e., frequently in junk messages and rarely in non-junk messages). TD ("term discriminability") provides a good indicator of junk messages by measuring the precision of the terms for the specific purpose of classifying junk messages.

TDTF computes the product of frequency of each term in the document (TF) with the term disriminability (TD).

Mail filters in popular mail programs like Eudora have always been able to filter messages based on the presence of specific keywords in the message body. One could, for example, establish a Eudora filter that automatically deletes any message containing the word "sex". In fact, we use this capability for processing the mail that a plugin implementing this invention classifies as junk. The plugin adds a unique keyword to the message to indicate that it is junk, and the user can set up a Eudora filter that redirects the message to a special mailbox, deletes it, or takes some other action on the message. The present invention is more powerful than the simple Boolean keyword search in that it uses an extended vocabulary, with or without term weights, to distinguish junk messages from non-junk messages. With the Eudora filters, it is an all-or-nothing affair. If the keyword is present, it is classified as junk. If the keyword is not present, the message slips through the filter. The present invention measures the degree to which a message should be classified as junk. There are many words, like "money", which are ambiguous as to whether the message is junk or not. The present invention counts the frequency of occurrence of such terms, along with other common warning signs of junk messages, to provide a qualitative measure of whether a message is junk or not.

Although TFIDF, Naïve Bayes, and similar methods have been used for filtering e-mail (see, for example, Jason Rennie's ifile system), they suffer from a sparse data problem. It is very hard for document similarity metrics like TFIDF and Naïve Bayes to classify documents when they have very few exemplars of the class. Such metrics need large quantities of data in order to work.

We address the sparse data problem by establishing a large, well-formulated query in advance by training on a large corpus of junk messages. Not only does this allow us

to accurately identify junk messages without relying on the user to compile and maintain their own corpus of junk messages, but it works immediately, right out of the box. The idea of preparing a well-formulated query for a specific filtering task in advance represents an improvement to the state of the art. It is not possible to do this for the user's own classification system, in general, but for a sharply focused and important problem like eliminating junk messages, it is easy and effective.

SUMMARY OF THE INVENTION

Briefly, according to this invention, there is provided a computer implemented method of filtering of junk messages by analyzing the content of the message instead of or in addition to using the message headers. This method involves document classification using a variety of information retrieval methods, but with unusually large queries. The term "queries", as used herein, refers to searches for terms in messages (or other documents) that match a list of terms (or lexicon). In this invention, a list of terms may include multiple word n-grams. The present invention uses very large queries (on the order of 250, 500 or 1,000 query terms or more in the lexicon) to achieve extremely high accuracy in classifying documents. The key is to pick topics for which a large set of exemplars is available so that the large queries can be constructed. Besides using the invention to filter junk messages, other possible applications include identifying job announcements, categorizing classified advertisements (e.g., "for sale" versus "wanted", real estate, automobiles and so on), appropriateness for children and other well-defined categories. The present invention may also be used to classify web pages and newsgroup postings in addition to e-mail. Since the categories are static but are of widespread interest, the time invested in constructing large queries will be worthwhile and can be invested by the software manufacturer instead of the end-user.

Junk mail, for example, is filtered by computing the sum of the product of the frequency of occurrence with the term weight for every term from the term lexicon that also appears in the message. The resulting sum is normalized by dividing the result by the total number of words (or the number of unique words) in the document. In other words, it is the dot product of the term frequency vector with the term weight vector perhaps normalized by document length. The key to the accuracy of this method is a large lexicon. This method permits alternate desired term weighting schemes.

According to a preferred method, the document or message is broken up into equal size chunks of the same number of words, with the score for the document taken as the maximum score for any chunk in the document. The last, odd-sized chunk may be merged into the previous chunk. Typical chunk sizes may be 50, 100 and 200 words.

According to one embodiment, the term weights are uniformly set equal to 1. According to another embodiment, a term's weight is its classification accuracy, as measured in a training corpus. Classification accuracy is the probability that the message is Junk given the Term is found in the message, that is, P(Junk | Term). The term weights are adjusted to occur above a minimum term weight (e.g., .1%), so that terms which are not present in the training corpus have non-zero term weights. In yet another embodiment, the term weights are the information gain, log(P(Term | Junk)). This embodiment makes use of the Naive Bayes method, but modified to allow the use of word n-grams (bigrams, trigrams, etc.) in addition to word unigrams.

A novel method disclosed herein uses word n-gram statistics (including unigram, bigram, trigram and mixed-length n-grams) on message content to identify junk messages. Another novel method disclosed herein involves using a product of term weights with term frequencies.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

The present invention uses a content-based method to identify the likelihood of a message being a junk message based on the content of the message itself. The language used in junk messages has characteristics that make it detectable. These methods offer a much higher accuracy than the prior art in correctly classifying messages as either junk or non-junk. The present invention has an accuracy that surpasses the effectiveness of header-based methods and is of sufficient accuracy to be used in stand-alone fashion to filter junk messages. However, there is no reason why it cannot be combined with header-based methods, and it is expected that this combination will be able to stop virtually all junk messages. Because the method is based on the content of the message with a rather fine-grained filter, the junk messages cannot be easily modified to bypass the filter.

The present invention automatically identifies whether a message, such as a piece of e-mail or newsgroup posting, is junk; marks it as junk; and either automatically discards the message or automatically files it in a junk mail folder (directory or subdirectory) for later review and disposition by the user (with the name of the folder designated either by the program or by the user).

The present invention includes a user-settable threshold that determines whether a message is classified as junk or not. If the message's bogosity score is above the threshold, it is classified as junk. Otherwise, it is classified as non-junk. The user can set the threshold lower to let no junk through but occasionally misclassify real messages as junk. The user can set the threshold higher to catch most, but not all, of the junk messages while not misclassifying any of the real mail or the user can set the threshold somewhere between the two thresholds.

This threshold may be set automatically to the value necessary to maximize the overall accuracy in classifying messages as junk or non-junk. Given a

collection of messages classified correctly and a set of misclassified messages, it is a straightforward process to find the threshold value that minimizes the number of classification errors. Since the number of messages classified as junk decreases as the threshold increases and the number of real messages classified as junk decreases as the threshold decreases, there is a threshold value that minimizes the number of classification errors. Common search methods, like hill-climbing and binary search, can be used to find it. This is similar to the methods we described for adjusting the term weights in the lexicon, but applies to the threshold value instead of the lexicon weights.

There are many phrases which are quite common in junk messages but significantly less common in legitimate correspondence. Examples include "credit card", "please pardon the intrusion", "make money fast", "extremely lucrative opportunity", "dear adult webmaster", "completely legal", "opportunity of a lifetime", "check or money order", "credit repair", "very lucrative", "limited time offer", and "to be removed". A lexicon of such phrases may be compiled through a combination of automated methods and human judgment.

In one embodiment of this invention, referred to herein as the "bogosity" method , one measures the degree to which the content of the message relies on a restricted lexicon of terms common in junk mail. This yields a "junk density" or bogosity figure. The higher this figure, the greater the degree to which the message uses the telltale signs of junk, and hence the greater the likelihood that the message is junk. Given a junk density threshold, the system can classify as junk any message with a bogosity score above the threshold.

The bogosity method breaks up the messages into, say, 100 word chunks, and counts the number of word n-grams (multiple word phrases) in each chunk which also appear in the lexicon of phrases that are indicative of junk messages.

The result is normalized by dividing it by the number of words in the chunk. The default chunk size can be set by the user. Typically, the chunk size will vary between 50 and 200. The bogosity score of the chunk with the highest bogosity score is used as the overall bogosity score of the message. The last chunk in the message may be less than the default chunk size. The bogosity method may ignore this chunk or merge it in with the previous chunk depending on the number of words in the chunk and the number of chunks in the message.

According to another embodiment, referred to herein as the TDTF method, weights are applied to each lexicon entry according to the Term Discriminability (classification accuracy) learned from a training corpus. Lexicon entries that are more indicative of junk will have higher weights than entries which are more ambiguous in nature. Negative weights are also permitted to allow the lexicon to include negative examples (e.g., good indicators of non-junk). This is the TDTF algorithm, where TD stands for term discriminability and TF stands for term frequency.

A variation on the embodiments described uses a library of example junk messages in case-based fashion. The idea is to use the exemplar messages as lexicons and to use an algorithm like bogosity to measure the similarity between the incoming e-mail and each of the messages in the library. If the similarity score for any junk message in the library with the incoming message exceeds a threshold, the incoming message would be classified as junk. This is similar in implementation, although somewhat different in conception, with the difference deriving from the use of the exemplar messages themselves as the lexicons and the use of many smaller lexicons (corresponding to each of the exemplar messages) instead of one large lexicon.

According to yet another embodiment of this invention, use is made of the Naïve Bayes statistical method that measures the information gain of classifying the messages using each word from the training corpus and

computes the overall likelihood of each message. For example, the top 20 words in the junk class sorted by log likelihood values are: money, report, business, order, orders, mail, e-mail, receive, free, send, credit, bulk, marketing, internet, program, cash, service, people, opportunity and product. This matches our intuitions about what terms are good indicators of junk messages. The benefits of Naïve Bayes are that it is a statistically well-founded technique which weights according to likelihood and incorporates notions of positive and negative weights by using separate scores for junk and non-junk and comparing the two.

A problem with Naïve Bayes is the assumption that words occur independently. For example, the word "report" may be a good indicator of junk mail (many pyramid schemes use this word), but it also filters out messages about progress reports. This problem is remedied by gathering statistics on word n-grams (e.g., word bigrams and trigrams) in addition to single words.

At a basic level, the bogosity, TDTF, and Naïve Bayes methods are similar in implementation. They each maintain a lexicon of terms (single words, word bigrams, word trigrams and word n-grams in general, as well as word n-grams with stop words removed) with weights associated with each term. For bogosity the weight is set equal to 1. For TDTF the weight is the trained classification accuracy (term discriminability) of the term, which is equivalent to the probability that the message is junk given the term,

P(Junk | Term).

For Naïve Bayes, the weight is the information gain, which is the logarithm of the probability of the term, given that the message is junk,

log (P(Term | Junk)).

Given these weights, the score for a document (or a chunk of a document) is the dot product (the sum of products, a linear combination of products) of the term

frequencies with the corresponding term weights, perhaps normalized by document length.

Various methods have been used on a corpus of junk and non-junk messages, computing the accuracy in classifying junk and non-junk, as well as the overall classification accuracy. It is important not only that the method identify junk, but also that it not mistakenly identify non-junk as junk. Those skilled in the art can quickly write a program for scanning the corpus of junk documents to develop the weights for terms found in the documents.

When the TDTF algorithm's weights are trained using different data than was used to construct the lexicon, some lexicon terms might not appear in the training data. This can happen when human judgment is used to add simple variations to the lexicon terms (e.g., adding a new term that corrects a spelling error in a lexicon term). The new term will not necessarily occur in the training data and so might be assigned to a score of 0. It is important to adjust the scores so that this term has a small non-zero value.

As noted previously, the junk accuracy of the heuristic (user not listed as a recipient) was about 50%, and the junk accuracy of blacklists was about 70%. The bogosity embodiment with a 0.20 threshold had a junk classification accuracy of about 90%, a non-junk classification accuracy of about 96% and an overall classification accuracy of about 95%. (Raising the threshold reduces the junk classification accuracy while increasing the non-junk classification accuracy. The 0.25 threshold seemed like a reasonable compromise.) The TDTF method with a threshold of 0.20 had junk, non-junk and overall classification accuracy scores of about 91%, 96% and 95%. Increasing the threshold to 0.25 reduced the junk accuracy to about 81% but increases the non-junk classification accuracy to 98%, with an overall accuracy of about 97%. The method using Naïve Bayes with unigrams had a junk classification accuracy of about 97%, non-junk about

96% and overall 96%. The method using Naïve Bayes with bigrams had a junk classification accuracy of about 98%, a real classification accuracy of about 98% and an overall classification accuracy of about 98%. Thus, the present invention represents a significant improvement to the state of the art.

Alternate implementations would involve several variations on the theme. For example, one implementation would train the lexicon on the user's own e-mail when the user installed the program. Another implementation would provide a ready-made lexicon and weights, and would allow the user to add new terms to the lexicon, delete terms from the lexicon and manually adjust the weights. Yet another implementation would also automatically adjust the weights when presented with new examples of junk and non-junk by small increments (for positive examples) and small decrements (for negative examples) for the terms found in the example. The increments and decrements would be computed using a variety of methods, such as gradient descent.

Prototypes of each of these methods have been implemented in Perl and C. It has been found it is quite useful in practice with Unix mail. It has been implemented as a plugin for the popular Windows and Macintosh mail program Eudora. The latest version also includes adjustable thresholds, whitelists and blacklists, and can highlight significant keywords in the e-mail message.

A copy of the PERL source code for a stand-alone version of bogosity and part of its lexicon follow. For an explanation of the PERL language, reference is made to Learning Perl, Second Edition, by Randal L. Schwartz and Tom Christiansen (O'Reilly & Associates, Inc. 1997).

SOURCE CODE FOR BOGOSITY.PL

```
$rootdir = "C:\\usr\\mkant\\Bogosity\\";
$mailfile = $ARGV[0];
$mailfile = "mail.txt" if (!$mailfile);
# the file of bogus words and phrases
$phrasefile = "bogosity.txt";
# number of words per chunky
$chunksize = $ARGV[1];
$chunksize = 200 if (!$chunksize);
# Let -ly and -est contribute to bogosity
$lyest = 1;
# For counting ! and ?
$maxrictus = 0;
$rictus = 0;
# Load the phrase file.
open(PHRASE, "$rootdir$phrasefile");
foreach $phrase (<PHRASE>) {
    chop $phrase;
    $phrases{"$phrase"} = 1;
}
close(PHRASE);
# process the mail
$maxbogosity = 0;
$wordcount = 0;
$bogosity = 0;
$prev = "";
$pprev = "";
$ppprev = "";
$pppprev = "";
open(MAIL, "$rootdir$mailfile");
foreach $line (<MAIL>) {
    chop $line;
                    if     ($line    !~     /^From:|^News
groups:|^Subject:|^Date:|^Organization:|^Lines:|^Message-I
D:|^References:|Mime-Version:|X-.*:|NNTP-Posting-Host:|^Pa
th:|Content-.*:/) {
     @lwords = split(/\s+/,$line);
     foreach $word (@lwords) {
         $lword = $word;
         $lword =~ s/\'s$//;
         $lword =~ s/\W$|\.$//;
         $lword =~ s/^\W|^\.//;
         $lword =~ tr/A-Z/a-z/;
         if (length("$word") < 25 &&
                              $word !~ /.+\@.+/i  &&  # was
ca,com,de,edu,gov,mil,net,org,uk,us
                 $word !~ /rec\./i &&
                 $word !~ /comp\./i &&
                 $word !~ /soc\./i &&
                 $word !~ /sci\./i &&
                 $word !~ /\.forsale/i &&
                 $word !~ /\.general/i &&
                 $word !~ /misc\./i &&
                 $word !~ /alt\./i &&
                 $word !~ /news\./i &&
```

```
                    $word !~ /<URL:/i &&
                    $word !~ /http:\/\//i &&
                    $word !~ /ftp:\/\//i &&
                    $word !~ /name\s*=/i &&
                    $word !~ /href\s*=/i &&
                    $word !~ /gopher:\/\//i &&
                $word !~ /^--------------/ &&
                    $lword !~ /^[\d\- (),\.\$]*$/ &&
                    $lword ne "") {
                        $wordcount++;
                if ($phrases{"$lword"} == 1) {
                    $bogosity++;
                } elsif ($lyest == 1 &&
                        ($lword =~ /ly$|est$/)) {
                    $bogosity++;
                }
                if ($phrases{"$prev $lword"} == 1) {
                    $bogosity++;
                }
                if ($phrases{"$pprev $prev $lword"} == 1) {
                    $bogosity++;
                }
                if ($phrases{"$pprev $pprev $prev $lword"} == 1)
    {
                    $bogosity++;
                }
                if  ($phrases{"$ppppprev  $pppprev  $ppprev  $prev
    $lword"} == 1) {
                    $bogosity++;
                }

                if ($word =~ /\?$|\!$/) {
                    $rictus++;
                }

                if ($wordcount >= $chunksize) {
                    if ($bogosity > $maxbogosity) {
                     $maxbogosity = $bogosity;
                    }
                    if ($rictus > $maxrictus) {
                     $maxrictus = $rictus;
                    }
                    $wordcount = 0;
                    $bogosity = 0;
                    $rictus = 0;
                }

                      $ppppprev = $pppprev;
                      $pppprev = $ppprev;
                      $ppprev = $pprev;
                      $pprev = $prev;
                      $prev = $lword;
            }
        }
      }
    }
    close(MAIL);
```

```
    printf "Maximum Bogosity: %.3f ($maxbogosity/$chunksize)\n",
    $maxbogosity/$chunksize;
    printf     "Maximum     Chunk     Rictus     (!?):     %.3f
    ($maxrictus/$chunksize)\n", $maxrictus/$chunksize;

 5                    BOGOSITY.TXT (partial)
    !!
    !!!
    !!!!
    !!!!!
10  !!!!!!
    !!!!!!!
    $
    $$
    $$$
15  $$$$
    $$$$$
    $$$$$$
    $$$$$$$
    $2.7 billion
20  $50,000
    $50,000 dollars or more
    $6.6 billion
    $70,000
    *this* mailing list
25  1,000
    1,000,000
    10,000
    100%
    100% committed
30  100% legal
    100% of the time
    100% satisfied
    100,000
    1000%
35  1302
    1342
    18 years old
    1st level
    1st time
40  200%
    2nd level
    3 level
    300%
    3rd level
45  4 level
    400%
    4th level
    500%
    5th level
50  8 level
    90-day limited warranty
    Four-level
    a brand new social security number
    a copy of
55  a couple of
```

a credit card
a deep breath
a different report
a few
a few hours
a few minutes
a large amount of money
a leading
a letter
a limited number
a list
a little bit
a little time
a lot
a lot easier
a lot more
a lot of
a lot of money
a lot of time
a mail box
a mailbox
a mailing list
a mailing list company
a mailing of
a miracle
a month
a must
a sign
a significant advantage
a sound way
a special program
a testimonial
a ton of money
a top leader
a total of
a total of perhaps
a variety of
ability
about to make
absolutely
absolutely convinced
absolutely free
absolutely guarantee
absolutely guaranteed
absolutely no credit check
absolutely no other fees
absolutely no risk
absolutely nothing
abuse
accept all credit cards
accept all major credit cards
accept american express
accept amex
accept cash
accept check
accept checks

```
accept credit cards
accept creditcards
accept major credit cards
accept master
accept mastercard
accept money orders
accept payment
accept personal checks
accept visa
accept visa/master
access fees
account executive
account number
account representative
acquiring e-mail lists
acquiring email lists
act fast
act now
action
activity level
ad
ad banner
ad below
ad campaign
ad length system
added bonus
additional income
address
address city
addressed
addresses
addresses accurately
```

The program flow can generally be described as follows. The lexicon file containing the words and phrases characteristic of junk mail, "bogosity.txt", and the file containing the mail, "mail.txt", are opened. A word is input from the mail.txt file and compared to the lexicon. If a match is found the score for that word (in this case always the same) is added to the raw score. The first word is kept so that it along with the next word can be compared to double-word phrases in the lexicon. Words and phrases (in this case up to five-word phrases) are compared to the lexicon and scored. When the maximum chunk size has been read and compared to the lexicon, the total score is divided by the chunk size. The next chunk is then analyzed. A running maximum score for the chunks of the message is kept and used as the score for the message. If the last chunk is

too short, it is merged with the next-to-last chunk or discarded. Finally, a line of text is added to the message to tag it as junk or not. Most mail programs have the capability of filing or discarding messages based upon this added line of text. This program is easily modified to implement the TDTF method and the Naive Bayes methods. The only difference is the use of different weights for terms in the lexicon.

Having thus defined our invention in the detail and particularity required by the Patent Laws, what is desired protected by Letters Patent is set forth in the following claims.

WE CLAIM:

1. A computer implemented method for filtering of junk messages comprising analyzing the content of the messages.

2. A computer implemented method for classification of a document as a junk message comprising analyzing the content of documents for the presence or absence of more than 250 words and/or multiple word n-grams.

3. A computer implemented method for classification of a document as a junk message comprising the steps of:

a) computing the sum of the product of the frequency of occurrence with an assigned term weight for every term and/or multiple word n-grams from a term lexicon that also appears in a document; and

b) assigning a score to the document based on the resulting sum.

4. The method according to claim 3, comprising the step of normalizing the resulting sum by dividing the result by the total number of words (or the number of unique words) in the document.

5. The method according to claim 3, wherein the document is broken up into equal sized chunks of the same number of words, with the score for the document as the maximum score for any chunk in the message.

6. The method according to claim 3, 4 or 5, comprising the further step of comparing the score assigned to the document to an adjustable threshold and classifying the document on the basis of that comparison.

7. The method according to claim 3, 4 or 5, wherein the term weights are uniformly set equal to 1.

8. The method according to claim 3, 4 or 5, wherein a term's weight is its classification accuracy P(Junk | Term), as measured in a training corpus.

9. The method according to claim 3, 4 or 5, wherein the term weights are the information gain, log(P(Term | Junk)) as measured in a training corpus.

10. The method according to claim 3, 4 or 5, wherein the term weights are supplied by the dependency tree algorithm.

11. The method according to claim 3, 4 or 5, with any monotonic modification of the weights.

12. The method according to claim 3, 4 or 5, wherein the lexicon is comprised of a plurality of lexicons and a score is assigned to the document based upon maximum score using any one of the plurality of lexicons.

13. The method according to claim 12, wherein the plurality of lexicons includes one or more junk messages.

14. The method according to claim 3, 4 or 5 applied to e-mail documents or the like, wherein message headers are compared with a blacklist to block messages that match header-based constraints.

15. The method according to claim 3, 4 or 5 applied to e-mail documents or the like, wherein message headers are compared with a whitelist to pass through messages that match header-based constraints.

16. The method according to claim 14, wherein only documents that are not blocked by the blacklist constraint are classified.

17. The method according to claim 15, wherein only documents that pass the whitelist constraint are classified.

18. The method according to claim 6 applied to e-mail documents or the like, wherein the user can set the threshold to let no junk mail through but occasionally misclassify a non-junk message as junk or the user can set the threshold to block most, but not all, junk messages while not misclassifying and non-junk messages or the threshold can be set somewhere therebetween.

19. The method according to claim 1, comprising a step for assigning a score to the document based on the content thereof which uses and falls with the likelihood that the message is junk and a step for comparing the score to a threshold to determine whether the message should be classified as junk.

20. The method according to claim 19, comprising the step for adjusting the threshold to control the balance between identifying junk messages and misclassifying non-junk messages.

21. The method according to claim 19, comprising a step for automatically setting the threshold to minimize all classification errors.

22. The method according to claim 3, 4 or 5, wherein the lexicon is derived from a training set of documents.

23. The method according to claim 22, wherein every term and/or multi-word n-gram in the lexicon has at least a minimum value.

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US99/24359

A. CLASSIFICATION OF SUBJECT MATTER
IPC(6) :G06F 13/00, 13/38, 17/00, 17/16, 17/21
US CL :Please See Extra Sheet.
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 709/203, 206, 207, 224, 238, 240; 707/500, 538; 379/93.01, 93.24, 100.08

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Roget's International Thesaurus , 5th Edition

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

Please See Extra Sheet.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y,E	US 5,999,932 A (PAUL) 07 DECEMBER 1999, ABSTRACT, FIGS. 1A, 3, 4, 4A, 6, COL. 2, LINES 1-60, COL. 5, LINES 5-67, COL. 6, LINES 1-40, COL. 8, LINES 17-67 AND COL. 9, LINES 1-67	1, 14-17,
Y	US 5,675,819 A (SCHUETZE) 07 OCTOBER 1997, FIGS. 3, 5-9, 12, 15 AND 16, COL. 3, LINES 28-34, COL. 15, LINES 13-64, COL. 17, LINES 13-67, COL. 18, LINES 35-67, COL. 19, LINES 1-67 AND COL. 20, LINES 1-6	2-5
Y	US 5,687,364 A (SAUND et al.) 11 NOVEMBER 1997, COL. 1, LINES 63-67, COL. 3, LINES 14-67, COL. 4, LINES 1-63, COL. 5, LINES 25-67, COL. 6, LINES 18-67, AND COL. 7, LINES 1-16	3, 6-13, 22 AND 23

[X] Further documents are listed in the continuation of Box C. [ ] See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E"	earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search 01 FEBRUARY 2000	Date of mailing of the international search report 25FEB2000
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer FRANK J. ASTA James R. Matthews Telephone No. (703) 305-3817

Form PCT/ISA/210 second sheet)(July 1992)*

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US99/24359

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y,P	US 5,905,863 A (KNOWLES et al.) 18 MAY 1999, FIG. 3, COL. 4, LINES 6-67, COL. 5, LINES 1-67, COL. 6, LINES 18-67 AND 1-4	2, 3, 6 AND 18-21
Y	US 5,619,648 A (CANALE et al.) 08 APRIL 1997, FIGS. 1-4, COL. 2, LINES 1-48, COL. 3, LINES 36-55,	1
A	US 5,826,022 A (NIELSEN) 20 OCTOBER 1998, ABSTRACT, COL. 2, LINES 59-66, COL. 6, LINES 33-38	1
Y	US 5,659,766 A (SAUND et al.) 19 AUGUST 1997, ABSTRACT, COL. 3, LINES 21-60, COL. 6, LINES 1-67	2-4
A	US 5,790,935 A (PAYTON) 04 AUGUST 1998, ABSTRACT	1
A	US 5,493,692 A (THEIMER et al.) 20 FEBRUARY 1996, ABSTRACT	
A	US 5,742,769 A (LEE et al.) 21 APRIL 1998, ABSTACT, COL. 1, LINES 36-67	1
A, P	US 5,832,212 A (CRAGUN et al.) 03 NOVEMBER 1998, ABSTRACT, COL. 1, LINES 1-67, COL. 2, LINES 15-49 , COL. 3, LINES 54-67 AND COL. 4, LINES 1-66	1 AND 4
A, P	US 5,963,965 A (VOGEL) 05 OCTOBER 1999, ABSTRACT	2-4
Y	MARCHIONINI, GARY, A CONCEPTUAL FRAMEWORK FOR TEXT FILTERING, ABSTRACT, PAGES 10, 15-1,8	2 3

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US99/24359

A. CLASSIFICATION OF SUBJECT MATTER:
US CL :

709/203, 206, 207, 224, 238, 240; 707/500, 538; 379/93.01, 93.24, 100.08

B. FIELDS SEARCHED
Electronic data bases consulted (Name of data base and where practicable terms used):

East, Internet, Internet Search Engine, Dialog
search terms: junk, email, documents, messages, algorithms, unwanted, unsolicited, weight, catergory, classify, score, rank, threshold, lexicon, frequency, analyze, measure